C-ROADS

C-ITS Security Requirements & Specifications

C-Roads platform, Working Group 2, Task Force 1

Version 0.7

12th May 2020

C-ROADS

Table of Content

Document history

Version	Date	Description, updates and changes	Status
0.5	25.10.2019	First draft for WG2 review	Draft
0.6	07.05.2020	Update following 1st review Integration of former report's annexes A & B	Draft

List of used abbreviations

AA	Authorization Authority
AT	Authorization Ticket
API	Application Programming Interface
CA	Certificate Authority
C-ITS	Cooperative ITS
CP	Certificate Policy
CPA	Certificate Policy Authority
CPS	Certificate Practice Statement
CPOC	C-ITS Point of Contact
CTL	Certificate Trust List
EA	Enrolment Authority
EC	Enrolment Certificate
EE	End Entity
ECTL	European Certificate Trust List
GDPR	General Data Protection Regulation
ITS	Intelligent Transport System
ITS-S	ITS Station
MS	Member State
OBU	On Board Unit
PKI	Public Key Infrastructure
SP	Security Policy
TBC	To Be Confirmed
TBD	To Be Defined
TF1	Task Force 1 – Security Aspects
TLM	Trust List Manager
TLS	Transport Layer Security - Internet Engineering Task Force (IETF) RFC 8446
WG2	Working Group 2

1 Introduction

The European ITS-Station architecture, outlined in EN 302 665, defines four ITS sub-systems; vehicle, roadside, personal, and central Standards are developed in a neutral and open way such that they include different options to allow diversion and future options to extend the standards later. To realize Interoperability among sub-systems many of these options need to be made specific. Profiles therefore describe the selected options and include additional specification when required to ensure the expected interoperability. Herein, the roadside sub-system profile is defined.

The Infrastructure Roadside Wi-Fi ITS-G5 System Profile, short called Roadside System Profile (RSP), defines a common base for the Wi-Fi ITS-G5 communication between roadside and vehicle. The communication directions derived from this are also known as I2V.

The profile provides descriptions, definitions and rules for all layers (Applications, Facilities, Networking & Transport and Access) of the ETSI ITS station reference architecture/ITS-S host. Management is included, but Security is out of scope. The understanding of the core infrastructure roadside system components is depicted in Table 1.

TABLE 1 - INFRASTRUCTURE ROADSIDE SYSTEM COMPONENTS

<table>
<tr><th>Layer</th><th colspan="2">Component</th><th>Tasks</th><th>Component</th></tr>
<tr><td>Applications</td><td colspan="2">Operational Specifications</td><td>Service definitions, and transmission principles, and triggering conditions</td><td rowspan="8">Management & Security</td></tr>
<tr><td rowspan="3">Facilities</td><td colspan="2">Positioning & Time
(incl. minimum data quality requirements)</td><td>Relevance Checking (C2C-CC White Paper on Positioning and Timing</td></tr>
<tr><td rowspan="2">Data and Message Content</td><td>CAM & DENM</td><td>Vehicle & infrastructure data provider (incl. minimum data quality requirements)</td></tr>
<tr><td>IVIM, SPATEM, MAPEM, etc.</td><td>Infrastructure data provider (incl. minimum data quality requirements)</td></tr>
<tr><td rowspan="3">Transport & Network</td><td>Transport</td><td>Basic transport protocol (BTP)</td><td>End-to-end, connection-less transport service</td></tr>
<tr><td rowspan="2">Network</td><td>Geo-Based Addressing</td><td>Future use</td></tr>
<tr><td>Geo-Routing Protocol</td><td>Future use</td></tr>
<tr><td>Access</td><td colspan="2">ETSI ITS-G5 European Profile Standard</td><td>Congestion Control</td></tr>
<tr><td colspan="4">IEEE 802.11p</td></tr>
</table>

2 Provisions

2.1 Verbal forms of the expression of provisions

In this document, the following verbal forms are used to indicate requirements:

Shall / Shall not

Recommendations shall be indicated by the verbal forms:

Should / Should not

Permissions shall be indicated by the verbal forms:

May / May not

Possibility and capability shall be indicated by the verbal forms:

Can / Cannot

Inevitability used to describe behavior of systems beyond of the scope of this deliverable shall be indicated by:

Will / Will not

Facts shall be indicated by the verbal forms:

Is / Is not

2.2 Provisions from referenced documents

Unless otherwise specified in the present document, the normative requirements included in the referenced documents supporting the required functionality of the ITS system shall apply. The verbal forms for the definition of provisions of referenced documents are defined either inside the document or generally by the SDO (standardisation organisation) or the organisation providing them. For example, normative requirements in ETSI documents are indicated by the verbal form “shall”.

2.3 Notation used to identify requirements

Some requirements of this document were directly copied, because they also apply on the security system. The identifier of source requirements is in brackets after the security requirement identifier (e.g. CP_001 for requirements taken from the Certificate Policy, RS_BSP_205 for C2C BSP).

2.4 Standards evolution

The standards chosen as specifications in this deliverable are evolving standards. This document selects specific versions of the underlying existing standards for concrete implementation.

3 Security requirements from reference documents

3.1 ETSI standards

Requirement *RS_SEC_001*

C-ITS stations (including central C-ITS stations) shall comply to **ETSI TS 103 097 V1.3.1** [1] and **ETSI TS 102 941 V1.3.1** [2] standards.

Note : a CA compliant with the EU Trust Model is used to issue certificates to backend C-ITS entities (i.e. central C-ITS stations) for message signing according to [1]. At the time of publication the specification only allows to sign on Geonetworking layer (according to ETSI EN 302 636-4-1 v1.3.1) but ongoing work in ETSI (ETSI Work Item DTR/ITS-00551) may allow signing on facilities layer).

Requirement *RS_SEC_002*

PKI shall comply to **ETSI TS 103 097 V1.3.1** [1] and **ETSI TS 102 941 V1.3.1** [2] standards.

NOTE : the compliance to these standards is a prerequisite to any security test conducted in C-Roads. C-ITS station and PKI should have passed basic interoperability tests described in ETSI TS 103 600 V1.1.1 [3]. These tests can be done during the test phases specific to each Member State.

3.2 Certificate Policy

A full compliance to the Certificate Policy is not required in C-Roads projects. However, the following CP requirements (for stations enrolment and authorization) shall be covered to ensure the proper level of security in nominal scenarios.

3.2.1 Stations enrolment

Compliance to ETSI Security standards requires C-ITS stations to be enrolled in a PKI. This procedure may be different from a PKI provider to another.

Stations operators should check with PKI providers the prerequisites to stations enrolment. At least, a canonical name (under a format to be defined) and a technical key under RFC 5480 SubjectPublicKeyInfo format are required.

Requirement *RS_SEC_003 (CP_040)*

Each C-ITS station shall be assigned two kinds of unique identifier:

- a canonical ID that is stored at the initial registration of the C-ITS station under the responsibility of the manufacturer. This shall contain a substring identifying the manufacturer or operator so that this identifier can be unique;
- a subject_name, which may be part of the C-ITS station's EC, under the responsibility of the EA.

Requirement *RS_SEC_004 (CP_124)*

Before a C-ITS station can request an EC certificate, the EE subscriber shall securely transmit the C-ITS station identifier information to the EA. The EA shall verify the request and in cases of positive verification register the C-ITS station information in its database and confirm this to the EE subscriber. This operation is done only once by the manufacturer or operator for each C-ITS station. Once a C-ITS station is registered by an EA, it may request a single EC certificate it needs at a time. The EA authenticates and verifies that the information in the EC certificate request is valid for a C-ITS station.

Requirement *RS_SEC_005 (CP_068)*

EE subjects of ECs shall authenticate themselves when requesting ECs by using their canonical private key for the initial authentication. The EA shall check the authentication using the canonical public key corresponding to the EE. The canonical public keys of the EEs are brought to the EA before the initial request is executed, by a secure channel between the C-ITS station manufacturer or operator and the EA.

Requirement *RS_SEC_006 (RS_BSP_456)*

The C-ITS station shall update its Enrolment Credential (EC) in advance before the expiration of its current valid EC, when the remaining validity duration of its Enrolment Credential is less than or equal to 3 Months.

3.2.2 Authorization

Requirement *RS_SEC_007*

AT key pair must be generated inside the HSM of the station. Stations shall use TS 102 941 v1.3.1 to request ATs.

Requirement *RS_SEC_008 (CP_042)*

The AA shall ensure that the pseudonymity of a C-ITS station is established by providing the C-ITS station with ATs that do not contain any names or information that may link the subject to its real identity according to TS 103 097 v1.3.1.

Requirement *RS_SEC_009 (CP_069)*

EE subjects of ATs shall authenticate themselves when requesting ATs by using their unique enrolment private key. The AA shall forward the signature to the EA for validation; the EA shall validate it and confirm the result to the AA. Request protocol according to TS 102 941 v1.3.1 shall be used by the EE and PKI.

Requirement *RS_SEC_010 (CP_075)*

The AA shall submit an authorisation validation request for each authorisation request it receives from an EE certificate subject according to TS 102 941 v1.3.1. The EA shall validate this request with respect to:

- the status of the EE at the EA;
- the validity of the signature;
- the requested ITS Application IDs (ITS-AID) and permissions;
- the status of service provision of the AA to the subscriber.

Requirement *RS_SEC_011 (CP_347, CP_348, RS_BSP_178)*

AT preloaded in the C-ITS station shall be compliant to [4].

3.3 Security Policy

A full compliance to the Security Policy [5] is not required in C-Roads projects.

3.4 CPOC & TLM

In C-Roads projects, implementation of the CPOC protocol [6] is not required. However, Root CA should comply to EU CCMS CPOC Protocol Annex 1.

3.5 Specific security requirements

3.5.1 Certificate format and validity times

Requirement *RS_SEC_012*

The certificates formats for CAs, ATs and ECs used for the C-Roads project are defined in ETSI TS 103 097 v1.3.1. Each C-ITS certificate is composed of several main fields:

- Version = 3,
- Signer Info = sha256AndDigest, sha384AndDigest or certificate,
- CRA CA Id = 0x000000,
- CRL Series = 0,
- Validity start with duration,
- App Permissions,
- Cert Issue Permissions only for CAs, and
- Signature (NIST or Brainpool with 256 Bit or Brainpool with 384 Bit).

Each certificate shall contain the complete list of explicit permissions..

Requirement *RS_SEC_013*

The field App Permissions contain the permission value in form of ITS-AID/PSID with SSP as long as specified for the respective ITS-AID. It needs to be considered that with ETSI TS 103 097 v1.3.1 there is no certificate type explicitly given in the certificates. The App Permissions field defines for which operations the key related to the certificate is allowed to be used. In addition, a root CA certificate is distinguished from an EA and AA certificate based on the signer info set to self.

The CA certificates shall contain Cert Issue Permissions which can be directly used by the CA to give them to the issued certificate into the App Permission field (EE Type set to app) or can be used in the enrolment process by an ITS station (EE Type set to enrol). It needs to be considered that with ETSI TS 103 097 v1.3.1 the end entity related to the EE Type is not necessarily the ITS station but can also be a CA if a specific permission end in the App Permission field of a CA certificate. Multiple elements of type Cert Issue Permission are used to distinguish between permissions that end in the directly issued certificates (permission with Minimum Chain Length = 1 and Chain Length Range = 0) or in certificates that are not directly issued by the certificate holder (permission with Minimum Chain Length > 1 and Chain Length Range = 0). The Chain Length Range shall be set in all CA certificates to 0.

The Root CA shall ensure that no CA certificate has a Cert Issue Permission with SSP bit set to 0 and the related Bitmask bit set to 0 for all SSP values defined as bit list. This is currently the case for all specified ITS-AID SSPs expect octet 1 to 3 of the IVI SSP according to ETSI TS 103 301 v1.2.1. If a certificate would contain a bit list permission in the Cert Issue Permission element with SSP bit = 0 and Bitmap bit = 0 then the certificate owner is not in possession of the respective permission but could issue the permission to a sub-certificate because Bitmap bit = 0 means that the SSP value is not considered in a chain permission check. This inconsistency needs to be prevented by the root CA and the sub-CAs.

Requirement *RS_SEC_014*

Validity times of certificates shall be compliant with the CP [4]

3.5.2 Cryptographic operations

Requirement *RS_SEC_015*

Cryptographic operations defined in the CP [4] sall be supported

---

3.5.3 Stations maximum permissions

Requirement *RS_SEC_016*

C-ITS stations shall use ATs with the maximum permissions defined in the following tables.

Requirement *RS_SEC_017*

The SSP format used in C-Roads project is of type opaque

Note : version bit is not included in the following tables.

CAM (PSID 36 - ETSI EN 302 637-2)

Octet Position	Bit Position	Permission Items	R-ITS-S (RSU)	VRO-ITS-S
1	0	CenDsrcTollingZone/ ProtectedCommunicationZonesRSU	1	0
1	1	publicTransport / publicTransportContainer	0	0
1	2	specialTransport / specialTransportContainer	0	0
1	3	dangerousGoods / dangerousGoodsContainer	0	0
1	4	roadwork / roadWorksContainerBasic	0	0
1	5	rescue / rescueContainer	0	0
1	6	emergency / emergencyContainer	0	0
1	7	safetyCar / safetyCarContainer	0	0
2	0	closedLanes / RoadworksContainerBasic	0	0
2	1	requestForRightOfWay / EmergencyContainer: EmergencyPriority	0	0
2	2	requestForFreeCrossingAtATrafficLight / EmergencyContainer: EmergencyPriority	0	0
2	3	noPassing / SafetyCarContainer: TrafficRule	0	0
2	4	noPassingForTrucks / SafetyCarContainer: TrafficRule	0	0
2	5	speedLimit / SafetyCarContainer	0	0
2	6	reserved for future usage	0	0
2	7	reserved for future usage	0	0

DENM (PSID 37 - ETSI EN 302 637-3)

Octet Position	Bit Position	CauseCodeType / Container	R-ITS-S (RSU)*	VRO-ITS-S*
1	0	trafficCondition(1)	1	1
1	1	accident(2)	1	1
1	2	roadworks(3)	1	1
1	3	adverseWeatherCondition-Adhesion(6)	1	1
1	4	hazardousLocation-SurfaceCondition(9)	1	0
1	5	hazardousLocation-ObstacleOnTheRoad(10)	1	1
1	6	hazardousLocation-AnimalOnTheRoad(11)	1	1
1	7	humanPresenceOnTheRoad(12)	1	1
2	0	wrongWayDriving(14)	1	0
2	1	rescueAndRecoveryWorkInProgress(15)	0	0
2	2	adverseWeatherCondition-ExtremeWeatherCondition(17)	1	1
2	3	adverseWeatherCondition-Visibility(18)	1	0
2	4	adverseWeatherCondition-Precipitation(19)	0	1
2	5	slowVehicle(26)	0	0
2	6	dangerousEndOfQueue(27)	1	1
2	7	vehicleBreakdown(91)	0	0
3	0	postCrash(92)	0	0
3	1	humanProblem(93)	0	0
3	2	stationaryVehicle(94)	1	1
3	3	emergencyVehicleApproaching(95)	0	0
3	4	hazardousLocation-DangerousCurve(96)	0	0
3	5	collisionRisk(97)	0	0
3	6	signalViolation(98)	0	0
3	7	dangerousSituation(99)	0	0

SPAT (PSID 137 - ETSI TS 103 301)

Octet Position	Bit Position	SPATEM data Item	R-ITS-S (RSU)	VRO-ITS-S
1	0	Signal Phase and Timing	1	0
1	1	Public transport prioritization status response	1	0
1	2	Manoeuvre assisting information	0	0

MAP (PSID 138 - ETSI TS 103 301)

Octet Position	Bit Position	RLT service SSP data Item	R-ITS-S (RSU)	VRO-ITS-S
1	0	Intersection geometry list allowed to transmit	1	0
1	1	Road geometry list allowed to transmit	1	0

IVI (PSID 139 - ETSI TS 103 301)

Octet Position	Bit Position	IVI data Item	R-ITS-S (RSU)	VRO-ITS-S
4	0	Vienna Convention Code for road sign	0	0
4	1	ISO/TS14823 traffic sign pictogram (danger warning)	1	0
4	2	ISO/TS14823 traffic sign pictogram (regulatory)	1	0
4	3	ISO/TS14823 traffic sign pictogram (informative)	1	0
4	4	ISO/TS14823 traffic sign pictogram (public facilities)	1	0
4	5	ISO/TS14823 traffic sign pictogram (ambient condition)	1	0
4	6	ISO/TS14823 traffic sign pictogram (road condition)	1	0
4	7	ITIS codes	0	0
5	0	Lane status	1	0
5	1	Road configuration container	1	0
5	2	Text container	1	0
5	3	Layout container	0	0
5	4	IVI status negation	0	0

TLC Request Service (PSID 140 - ETSI TS 103 301)

To be completed

3.5.4 Certificate Authorities maximum permissions

The Service Specific Permission (SSP) is a field that indicates specific sets of permissions within the overall permissions indicated by the PSID. For example, there may be an SSP value associated with the PSID for CAM that indicates that the sender is entitled to send CAMs for a specific vehicle role. SSPs are used in certificate requests (get EC and get AT) and during initialization phase.

In the following tables (examples), the value ‘x’ in Table 2 above shall contain the country code according to ISO 14816 and the value ‘y’ the provider ID which is defined in ETSI TS 103 301 V1.3.1 but currently not assigned to specific values. All ‘x’ and ‘y’ values may be set to 0 to allow supporting all possible country codes and provider IDs. For these three specific SSP bytes the related bitmask value might be set to 0.

Note : version bit is included in the following tables.

App Permissions		sspValue (hex)		sspValue (binary)	
624	CTL	0138		0000 0001 0011 1000	
622	CRL	01		0000 0001	
Explicit cert issue permissions with minimum chain length = 1 (default), chain length range = 0 (default) and end entity type = app (default)					
ITS-AID		**sspValue (hex)**	**Bitmask (hex)**	**sspValue (binary)**	**Bitmask (binary)**
623	SCR	013E	FFC1	0000 0001 0011 1110	1111 1111 1100 0001
Explicit cert issue permissions with minimum chain length = 2, chain length range = 0 (default) and end entity type = app (default)					
ITS-AID		**sspValue (hex)**	**Bitmask (hex)**	**sspValue (binary)**	**Bitmask (binary)**

36	CAM	01FFFC	FF0003	0000 0001 1111 1111 1111 1100	1111 1111 0000 0000 0000 0011
37	DENM	01FFFFFF	FF000000	0000 0001 1111 1111 1111 1111 1111 1111	1111 1111 0000 0000 0000 0000 0000 0000
137	TLM	01E0	FF1F	0000 0001 1110 0000	1111 1111 0001 1111
138	RLT	01C0	FF3F	0000 0001 1100 0000	1111 1111 0011 1111
139	IVI	01xxxyyyFFF8	FF0000000007	0000 0001 xxxx xxxx xxyy yyyy yyyy yyyy 1111 1111 1111 1000	1111 1111 0000 0000 0000 0000 0000 0000 0000 0000 0000 0111
140	TLC Request Service	02FFFFE0	FF00001F	0000 0010 1111 1111 1111 1111 1110 0000	1111 1111 0000 0000 0000 0000 0001 1111
141	GN_MGMT				
623	SCR	01C0	FF3F	0000 0001 1100 0000	1111 1111 0011 1111
637	TLC Status Service	01	FF	0000 0001	1111 1111

Table 2 - Specification of maximum permissions contained in a Root certificate

The EA contains one cert issue permission element with EE Type = app that is given to the EC certificate into the app permission field. An EA with the permissions listed in the following table is permitted to issue EC certificates with all possible permissions.

App Permissions		**sspValue (hex)**		**sspValue (binary)**	
623	SCR	010E		0000 0001 0000 1110	
Explicit cert issue permissions with minimum chain length = 1 (default), chain length range = 0 (default) and end entity type = app (default)					
ITS-AID		**sspValue (hex)**	**Bitmask (hex)**	**sspValue (binary)**	**Bitmask (binary)**
623	SCR	01C0	FF3F	0000 0001 1100 0000	1111 1111 0011 1111

Table 3 - Specification of permissions contained in a EA certificate

In addition, the EA and the Root CA certificates may contain in another cert issue permission element with EE Type = enrol all permissions that can be assigned to an ITS station registration. If the EA certificate would not contain the enrol permissions the root CA could not limit the permissions of an EA, e.g. one EA is allowed to register stations with extended permissions and another EA is allowed to handle only normal private stations.

App Permissions		**sspValue (hex)**		**sspValue (binary)**	
623	SCR	0132		0000 0001 0011 0010	
Explicit cert issue permissions with minimum chain length = 1 (default), chain length range = 0 (default) and end entity type = app (default)					
Issue Permissions		**sspValue (hex)**	**Bitmask (hex)**	**sspValue (binary)**	**Bitmask (binary)**
36	CAM	01FFFC	FF0003	0000 0001 1111 1111 1111 1100	1111 1111 0000 0000 0000 0011
37	DENM	01FFFFFF	FF000000	0000 0001 1111 1111 1111 1111 1111 1111	1111 1111 0000 0000 0000 0000 0000 0000

137	TLM	01E0	FF1F	0000 0001 1110 0000	1111 1111 0001 1111
138	RLT	01C0	FF3F	0000 0001 1100 0000	1111 1111 0011 1111
139	IVI	01xxx000FFF8	FF0000000007	0000 0001 xxxx xxxx xx00 0000 0000 0000 1111 1111 1111 1000	1111 1111 0000 0000 0000 0000 0000 0000 0000 0000 0000 0111
140	TLC Request Service	01FFFFE0	FF00001F	0000 0001 1111 1111 1111 1111 1110 0000	1111 1111 0000 0000 0000 0000 0001 1111
141	GN_MGMT				
637	TLC Status Service	01	FF	0000 0001	1111 1111

Table 4 - Specification of maximum permissions contained in a AA certificate

App Permissions		sspValue (hex)	sspValue (binary)
623	SCR	01C0	0000 0001 1100 0000

Table 5 - Specification of permissions contained in a EC certificate

3.5.5 Security initialisation

Requirement *RS_SEC_018*

Receiving C-ITS stations shall operate the initial verifications as described in Annex A - Security initialisation steps.

3.5.6 Message signature

This section also applies to hybrid Basic Interface.

Requirement *RS_SEC_019 (RS_BSP_160)*

The C-ITS station shall use one end-to-end security header per message in accordance with [1]

Requirement *RS_SEC_020 (RS_BSP_407)*

The signature shall be generated using a private key corresponding to a valid AT in accordance with clause 7.2.1 in [1].

Note : C2C (*RS_BSP_170*) recommends using digital signatures and certificates based on ECDSA-256 using the elliptic curve NIST P-256.

Requirement *RS_SEC_021 (RS_BSP_182)*

All addresses and identifiers of mobile stations transmitted shall be changed when the AT is changed according to section 6.5 of [11].

3.5.7 Verification of message signature

The Certificate Policy [4] requires the update of C-ITS stations with the ECTL and CRLs within a week of their publication.

C-ITS stations should update the ECTL and CRLs in a higher frequency. Especialy, it is recommended for RSUs to be closer to a daily frequency.

Requirement *RS_SEC_022 (RS_BSP_163)*

At each reception of message, C-ITS stations shall operate the signature verifications as described in Annex B - Signature verification steps. It is based on the section *5.2.3.2.1a Signature verification of IEEE 1609-2.*

Requirement *RS_SEC_023 (RS_BSP_164)*

The C-ITS station shall forward only verified messages.

3.5.8 Logging

In case of failure during initialisation or signature verification, the C-ITS station should log the reason of the verification error

Logs should at least include the type of error (e.g. invalid certificate, permissions mismatch) and the targeted element (e.g. TLM, ECTL, RCA, AA, AT).

Non exhaustive list of errors examples:

- Invalid TLM certificate
- Non trusted RCA
- Revoked AA
- RCA-AA permission mismatch
- Invalid message signature

3.6 Hybrid related requirements

The following requirements apply to C-ITS message exchanged on the Basic Interface as defined by TF4.

Requirement *RS_SEC_024 (HYB_017)*

C-ITS actors shall ensure the integrity of the information they exchange.

Requirement *RS_SEC_025 (HYB_040)*

According to the European C-ITS Certificate Policy [4], individual messages transmitted on the Basic Interface shall be signed according to ETSI TS 103 097 [1].

Requirement *RS_SEC_026 (HYB_041)*

The sending ITS station shall be responsible for the signature and the timestamping of the message.

Requirement *RS_SEC_027 (HYB_042)*

No signature change during message transport from the original sender to the final receiver shall be allowed.

Requirement *RS_SEC_028 (HYB_047)*

When transmitting C-ITS messages via different channels (technologies/ networks) all specific ETSI certificates and ID´s created with them shall be preserved to guarantee message authenticity.

3.7 Other vehicle stations requirements

Requirement *RS_SEC_029 (RS_BSP_158)*

A vehicle C-ITS station (e.g. for road operator vehicle) shall be securely linked to one specific vehicle. Where the vehicle C-ITS station is powered, it shall verify that it is operating in the vehicle with which it has been securely linked. If such correct functioning condition cannot be verified, the C-ITS station shall be deactivated, preventing it from sending messages (i.e. deactivate at least the radio transmission level of the C-ITS station).

Note: Securely linked means paired in the factory or in an authorized repair shop.

Requirement *RS_SEC_030 (RS_BSP_181)*

If the C2C-CC Basic System detects a collision of the least significant 32 bit of the "Certificate digest" / "hashedId8" with the "Certificate digest" / "hashedId8" of another ITS station (or C2CCC Basic System), it shall initiate a change of its authorization ticket if the certificate corresponding to the other "Certificate digest" / "hashedId8" is valid, if no such collision has triggered the current authorization ticket is used.

Requirement *RS_SEC_031 (RS_BSP_185)*

Facilities layer shall clear the own station's path history cache (used to fill into new messages) when the security entity changes its authorization ticket identity.

Requirement *RS_SEC_032 (RS_BSP_184)*

Applications shall be able to block the authorization ticket change indefinitely, if the vehicle does not move, i.e. PathPoint position information does not change. In other cases, applications shall only be able to block it for at most ***XXX***.

Exception:

- validity of the authorization ticket expired;
- collision of "Certificate digest" / "hashedId8".

4 References

All normative references within a standard referenced here are automatically included and will not be listed separately.

Only if a normative reference is out of date because a newer version of the reference standard is supported, the newer reference is listed and marked accordingly.

Table 6 Table of normative key references

#	Reference
[1]	ETSI TS 103 097 V1.3.1: ITS Security - Security header and certificate formats
[2]	ETSI TS 102 941 V1.3.1: ITS Security - Trust and Privacy Management
[3]	ETSI TS 103 600 V1.1.1: ITS Testing - Interoperability test specifications for security
[4]	Certificate Policy (ANNEX 3 to the Commission Delegated Regulation supplementing Directive 2010/40/EU of the European Parliament and of the Council with regard to the deployment and operational use of cooperative intelligent transport systems) – March 2019
[5]	Security Policy (ANNEX 4 to the Commission Delegated Regulation supplementing Directive 2010/40/EU of the European Parliament and of the Council with regard to the deployment and operational use of cooperative intelligent transport systems) – March 2019
[6]	C-ITS Point of Contact (CPOC) Protocol Release 1.1 (Draft)
[8]	20190827_TF1_Security_report_v1.7
[9]	TF4 BI report (version?)
[10]	C2CCC_RS_2037_Profile
[11]	ETSI TS 102 940 V1.3.1: ITS Security - ITS communications security architecture and security management

Annex A - Security initialisation steps

The TLM certificate is provided to the ITS-S during the initialization phase which is assumed to be valid and trustworthy.

- The ITS-S verifies that the issuer of the TLM certificate is set to self.
- The ITS-S verifies that the signature in the TLM certificate can be successfully verified with the public verification key provided in the TLM certificate.
- The ITS-S verifies that no Cert Issue Permissions are present in the TLM certificate.
- The ITS-S verifies that the start time of the TLM certificate is before the end time of the TLM certificate.
- The ITS-S verifies that the current time is equal to or after the start time of the TLM certificate.
- The ITS-S verifies that the current time is equal to or before the end time of the TLM certificate.

The ECTL is provided to the ITS-S during the initialization phase and is updated periodically according to the European C-ITS Certificate Policy release 1.1.

The home RCA certificate ID is provided to the ITS-S during the initialization phase. The ITS-S extracts the RCA certificate from the ECTL by using the ID. Alternatively, if the RCA certificate itself is provided to the ITS-S then the ITS-S must ensure that an exact copy of the RCA certificate is listed on the ECTL.

- The ITS-S verifies that the issuer of the RCA certificate is set to self.
- The ITS-S verifies that the signature in the RCA certificate can be successfully verified with the public verification key provided in the RCA certificate.
- The ITS-S verifies that the start time of the RCA certificate is before the end time of the RCA certificate.
- The ITS-S verifies that the current time is equal to or after the start time of the RCA certificate.
- The ITS-S verifies that the current time is equal to or before the end time of the RCA certificate.

The RCA-CTL is available in an RCA repository of each PKI operator.

The CRL is available in the RCA repository of each PKI operator and the ITS-S is equipped with a valid CRL of each RCA listed on the ECTL.

- For each Root CA Entry on the ECTL a DC Entry should be available on the ECTL, i.e. each RCA HashedId8 should be listed in at least one DC Entry Cert element.
- The operator of the ITS-S or the ITS-S itself connects periodically (i.e. every 48h) to all required DC URLs to download the CRL of each Root CA listed on the ECTL.
- The ITS-S verifies that the signer of the CRL is listed on the ECTL as RCA.
- The ITS-S verifies that the CRL signature can be successfully verified with the public verification key provided in the RCA certificate.
- The ITS-S verifies that the PSID 622 with SSP value 0x01 is listed in the App Permission element of the RCA certificate (see Table 2 - Specification of maximum permissions contained in a Root certificate).
- The ITS-S verifies that the start time of the RCA certificate is equal to or before the time given in the CRL for this update.
- The ITS-S verifies that the end time of the RCA certificate is equal to or after the time given in the CRL for next update.

- The ITS-S verifies that the start time of the RCA certificate is before the end time of the RCA certificate.
- The ITS-S verifies that the time given in the CRL for this update is before the time given in the CRL for next update.
- The ITS-S verifies that the current time is equal to or after the time given in the CRL for this update.
- The ITS-S verifies that the current time is equal to or before the time given in the CRL for next update.

The ITS-S shall verify that the ECTL is signed by the TLM.

- The ITS-S shall check the CTL format and the CTL sequence. If the format is set to FullCtl the ITS-S shall ensure that the latest ECTL (highest sequence number) is present. If the format is set to DeltaCtl the ITS-S shall ensure that all delta ECTLs with lower sequence numbers down to the last full ECTL (or sequence number = 0 if no full ECTL was processed previously) are present and will be checked with the following steps.
- The ITS-S verifies that the signer of the ECTL is the trusted and verified TLM certificate.
- The ITS-S verifies that the ECTL signature can be successfully verified with the public verification key provided in the TLM certificate.
- The ITS-S verifies that the PSID 624 is set in the ECTL header information.
- The ITS-S verifies that the PSID 624 with SSP value 0x01C8 is listed in the App Permission element of the TLM certificate.
- The ITS-S verifies that the start time of the TLM certificate is equal to or before the generation time given in the ECTL header information.
- The ITS-S verifies that the end time of the TLM certificate is equal to or after the time given in the ECTL for next update.
- The ITS-S verifies that the start time of the TLM certificate is before the end time of the TLM certificate.
- The ITS-S verifies that the generation time given in the ECTL header is before the time given in the ECTL for next update.
- The ITS-S verifies that the current time is equal to or after the generation time in the ECTL header information. The ITS-S verifies that the current time is equal to or before the time given in the ECTL for next update.

The ITS-S shall verify that the TLM link certificate is valid if present on the ECTL.

- The ITS-S ensures that the old TLM certificate is present, as well as the new TLM certificate and the TLM link certificate. All three TLM certificates should be given in the ECTL. The affiliation of the three certificates might be given by using the same name in the certificates.
- The ITS-S verifies the validity of the old and the new self-signed TLM certificates based on the steps above.
- The ITS-S verifies that the issuer of the TLM link certificate is set to the old TLM certificate by using the ID or certificate.
- The ITS-S verifies that the signature in the TLM link certificate can be successfully verified with the public verification key provided in the old TLM certificate.
- The ITS-S verifies that the signature in the new TLM certificate can be successfully verified with the public verification key provided in the new TLM certificate.
- The ITS-S verifies that all PSID with related SSP values in the App Permission element of the TLM link certificate are equal to the App Permission element of the new TLM certificate.
- The ITS-S verifies that the verificationKey present in the TLM link certificate matches exactly with the verificationKey in the new TLM certificate.
- The ITS-S verifies that the start time of the TLM link certificate is equal to the start time of the new TLM certificate.

- The ITS-S verifies that the start time of the old TLM certificate is before the start time of the TLM link certificate.
- The ITS-S verifies that the end time of the old TLM certificate is equal to the end time of the TLM link certificate.
- The ITS-S verifies that the start time of the TLM link certificate is before the end time of the TLM link certificate.
- If the TLM link certificate contains a region restriction the ITS-S verifies that this is equal to the region restriction of the new TLM certificate.
- If the TLM link certificate contains an assurance level the ITS-S verifies that this is equal to the assurance level contained in the new TLM certificate.

The ITS-S shall verify that the home RCA link certificate is valid if present on the ECTL.

- The ITS-S ensures that the old RCA certificate is present, as well as the new RCA certificate and the RCA link certificate. All three RCA certificates might be given in the ECTL. The affiliation of the three certificates might be given by using the same name in the certificates.
- The ITS-S verifies the validity of the old and the new self-signed RCA certificates based on the steps above.
- The ITS-S verifies that the issuer of the RCA link certificate is set to the old RCA certificate by using the ID or certificate.
- The ITS-S verifies that the signature in the RCA link certificate can be successfully verified with the public verification key provided in the old RCA certificate.
- The ITS-S verifies that the signature in the new RCA certificate can be successfully verified with the public verification key provided in the new RCA certificate.
- The ITS-S verifies that all PSID with related SSP values in the App Permission element of the RCA link certificate are equal to the App Permission element of the new RCA certificate.
- The ITS-S verifies that the certIssuePermissions and the verificationKey present in the RCA link certificate match exactly with the certIssuePermissions and the verificationKey in the new RCA certificate.
- NOTE: App Permission and Cert Issue Permission might change between old and new RCA certificate. A change shall be communicated and agreed according to the C-ITS CP [2], i.e. section 4.1.2.1 before the RCA certificate is added to the ECTL.
- The ITS-S verifies that the start time of the RCA link certificate is equal to the start time of the new RCA certificate.
- The ITS-S verifies that the start time of the old RCA certificate is before the start time of the RCA link certificate.
- The ITS-S verifies that the end time of the old RCA certificate is equal to the end time of the RCA link certificate.
- The ITS-S verifies that the start time of the RCA link certificate is before the end time of the RCA link certificate.
- If the RCA link certificate contains a region restriction the ITS-S verifies that this is equal to the region restriction of the new RCA certificate.
- If the RCA link certificate contains an assurance level the ITS-S verifies that this is equal to the assurance level contained in the new RCA certificate.

The ITS-S shall verify that the RCA-CTL is signed by the home RCA.

- The ITS-S shall check the CTL format and the CTL sequence number. If the format is set to FullCtl the ITS-S shall ensure that the latest RCA-CTL (highest sequence number) is present. If the format is set to DeltaCtl the ITS-S shall ensure that all delta RCA-CTLs with lower sequence numbers down to the last full RCA-CTL (or sequence number = 0 if no full RCA-CTL was processed previously) are present and will be checked with the following steps.
- The ITS-S verifies that the signer of the RCA-CTL is the home RCA, which is listed on the ECTL.
- The ITS-S verifies that the RCA-CTL signature can be successfully verified with the public verification key provided in the RCA certificate.

- The ITS-S verifies that the PSID 624 is set in the RCA-CTL header information.
- The ITS-S verifies that the PSID 624 with SSP value 0x0138 is listed in the App Permission element of the RCA certificate.
- The ITS-S verifies that the start time of the RCA certificate is equal to or before the generation time given in the RCA-CTL header information.
- The ITS-S verifies that the end time of the RCA certificate is equal to or after the time given in the RCA-CTL for next update.
- The ITS-S verifies that the start time of the RCA certificate is before the end time of the RCA certificate.
- The ITS-S verifies that the generation time given in the RCA-CTL header is before the time given in the RCA-CTL for next update.
- The ITS-S verifies that the current time is equal to or after the generation time in the RCA-CTL header information.
- The ITS-S verifies that the current time is equal to or before the time given in the RCA-CTL for next update.

Annex B - Signature verification steps

All RCA certificates contained on the ECTL are stored as trust anchor in a secure way in order to prevent unauthorized modification or exchange, e.g. inside the HSM.

Step (1): The ITS-S receives a secured message and verifies the message signature with the associated AT certificate.

- The ITS-S verifies that the signer in the security header info matches with the AT certificate.
- The ITS-S verifies that the signature can be successfully verified with the public verification key provided in the AT certificate.
- The ITS-S verifies that the PSID in the security header info is listed in the App Permission element of the AT certificate.
- The ITS-S verifies that required SSP bits are set to 1 in the signer AT certificate if the received message contains the respective container or element. The content of the message payload (e.g. existence of an emergencyContainer) need to be compared with the SSP bits in the signer AT certificate.
- The ITS-S verifies that the start time of the AT certificate is equal to or before the generation time in the security header info.
- The ITS-S verifies that the end time of the AT certificate is equal to or after the optionally contained expiration time in the security header info.
- The ITS-S verifies that the start time of the AT certificate is before the end time of the AT certificate.
- The ITS-S verifies that the generation time in the security header info is before the optionally contained expiration time in the security header info.
- The ITS-S verifies that the transmission location is within the optionally contained region restriction of the AT certificate.

Step (2): The ITS-S verifies that the AT certificate is issued by an AA with a valid certificate (e.g.: presence of the right PSID list, time start and duration…). The AA certificate may be retrieved either from a V2X secured exchange (requestedCertificate) or from a RCA-CTL.

- The ITS-S verifies that the issuer in the AT certificate matches with the AA certificate.
- The ITS-S verifies that the issuer signature in the AT certificate can be successfully verified with the public verification key provided in the AA certificate.
- The ITS-S verifies that PSID 623 with SSP bit 2 and 3 are set in the App Permission element of the AA certificate.
- The ITS-S verifies that all PSID with related SSP values in the App Permission element of the AT certificate are listed in one of the Cert Issue Permission elements of the AA with appropriate chain length (e.g. Min Chain Length set to 1 and Chain Length Range set to 0) and EE Type set to App.
    - Check 1: For each SSP Bitmask bit set to 1 in the AA Cert Issue Permission the related SSP bit value in the AA certificate need to be set in the AT certificate App Permission.

Issuer SSP bit:	0*	0*	0	0	1	1	1	1
Issuer bitmask bit:	0*	0*	1	1	0	0	1	1
Certificate App SSP bit:	0	1	0	1	0	1	0	1
Result of check at ITS-S	OK	NOK	OK	NOK	OK	OK	NOK	OK
Check algorithm that is used to get the result			1	1			1	1

Figure 1: AA - AT app permission check

*NOTE: The configuration SSP = 0 and Bitmask = 0 should not appear in CA certificates which should be enforced by the TLM, Root CA and AA.

- The ITS-S verifies that the start time of the AA certificate is equal to or before the start time of the AT certificate.
- The ITS-S verifies that the end time of the AA certificate is equal to or after the end time of the AT certificate.
- The ITS-S verifies that the start time of the AA certificate is before the end time of the AA certificate.
- The ITS-S verifies that the start time of the AT certificate is before the end time of the AT certificate.
- If the AT certificate contains a region restriction the ITS-S verifies that this is within the region restriction of the AA certificate.
- If the AT certificate contains an assurance level the ITS-S verifies that this is equal to or smaller the assurance level contained in the AA certificate.
- The ITS-S verifies that a valid CRL is available and that the HashedID8 of the AA is not listed on this CRL.

Step (3): The ITS-S verifies that the AA certificate is issued by RCA.

- The ITS-S verifies that the issuer in the AA certificate matches with a RCA certificate listed in the ECTL.
- The ITS-S verifies that the issuer signature in the AA certificate can be successfully verified with the public verification key provided in the RCA certificate.
- The ITS-S verifies that all PSID with related SSP values in the App Permission element of the AA certificate are listed in one of the Cert Issue Permission elements of the RCA with appropriate chain length (e.g. Min Chain Length set to 1 and Chain Length Range set to 0 in RCA certificate) and EE Type set to App.
    - Check 1: For each SSP Bitmask bit set to 1 in the RCA Cert Issue Permission the related SSP bit value in the RCA certificate need to be set in the AA certificate App Permission.

Issuer SSP bit:	0*	0*	0	0	1	1	1	1
Issuer bitmask bit:	0*	0*	1	1	0	0	1	1
Certificate App SSP bit:	0	1	0	1	0	1	0	1
Result of check at ITS-S	OK	NOK	OK	NOK	OK	OK	NOK	OK
Check algorithm that is used to get the result			1	1			1	1

Figure 2: RCA - AA app permission check

**NOTE: The configuration SSP = 0 and Bitmask = 0 should not appear in CA certificates which should be enforced by the TLM, Root CA and AA.*

- The ITS-S verifies that all PSID with related SSP values in the Cert Issue Permission element of the AA certificate are listed in one of the Cert Issue Permission elements of the RCA with appropriate chain length where the chain length is decreased by one (e.g. Min Chain Length set to 2 and Chain Length Range set to 0 in RCA certificate) and EE Type set to App.
    - Check 1: For each SSP Bitmask bit set to 1 in the RCA Cert Issue Permission the related SSP bit value in the RCA certificate need to be set in the AA certificate Cert Issue SSP.
    - Check 2: For each SSP Bitmask bit set to 1 in the RCA Cert Issue Permission this Bitmask value in the RCA certificate need to be set in the related AA certificate Cert Issue SSP Bitmask value.

Figure 3: RCA - AA certificate issuer permission Check

Root SSP bit:	0*	0*	0*	0*	0	0	0	0	1	1	1	1	1	1	1	1
Root bitmask bit:	0*	0*	0*	0*	1	1	1	1	0	0	0	0	1	1	1	1
AA SSP bit:	0	0	1	1	0*	0	1	1	0*	0	1	1	0*	0	1	1
AA bitmask bit:	0	1	0	1	0*	1	0	1	0*	1	0	1	0*	1	0	1
Result of check at ITS-S	OK	OK	NOK	NOK	NOK	OK	NOK	NOK	NOK	OK	OK	OK	NOK	NOK	NOK	OK
Check algorithm that is used to get the result					2	1,2	1,2	1					1,2	1	2	1,2

**NOTE: The configuration SSP = 0 and Bitmask = 0 should not appear in CA certificates which should be enforced by the TLM, Root CA and AA.*

- The ITS-S verifies that the start time of the RCA certificate is equal to or before the start time of the AA certificate.
- The ITS-S verifies that the end time of the RCA certificate is equal to or after the end time of the AA certificate.
- The ITS-S verifies that the start time of the RCA certificate is before the end time of the RCA certificate.
- The ITS-S verifies that the start time of the AA certificate is before the end time of the AA certificate.
- If the AA certificate contains a region restriction the ITS-S verifies that this is within the region restriction of the RCA certificate.
- If the AA certificate contains an assurance level the ITS-S verifies that this is equal to or smaller the assurance level contained in the RCA certificate.
- The ITS-S verifies that a valid CRL is available and that the HashedID8 of the RCA is not listed on this CRL.

Step (4): The ITS-S verifies the RCA certificate

- The ITS-S verifies that the issuer of the RCA certificate is set to self.
- The ITS-S verifies that the signature in the RCA certificate can be successfully verified with the public verification key provided in the RCA certificate.
- The ITS-S verifies that the start time of the RCA certificate is before the end time of the RCA certificate.
- The ITS-S verifies that the current time is equal to or after the start time of the RCA certificate.
- The ITS-S verifies that the current time is equal to or before the end time of the RCA certificate.

At every step of this procedure, if one verification fails, then the signed message must be considered as invalid and must be rejected by the ITS-S.

Post-conditions

The ITS-S has verified the integrity of the received message by validating the signature of the secured message as well as its authenticity by validating the certificate trust chain. The authorization of the message sender needs to be checked by the application which verifies that the required ITS-AID and SSP values are set in the sender's AT certificate.

Annex C - Certificate examples

The following decoded example certificates illustrate the contents of CA certificates used in C-Roads.

Example of Root CA certificate

```
EtsiTs103097Certificate ::= {
  version 3,
  type explicit,
  issuer self : sha256,
  toBeSigned {
    id name : "BSI V2X Pilot PKI Root",
    cracaId '000000'H,
    crlSeries 0,
    validityPeriod {
      start 477215871,
      duration hours : 17544
    },
    appPermissions {
      {
        psid 624,
        ssp bitmapSsp : '0138'H
      },
      {
        psid 622,
        ssp bitmapSsp : '01'H
      }
    },
    certIssuePermissions {
      {
        subjectPermissions explicit : {
          {
            psid 36,
            sspRange bitmapSspRange : {
              sspValue '01FFFC'H,
              sspBitmask 'FF0003'H
            }
          },
          {
            psid 37,
            sspRange bitmapSspRange : {
              sspValue '01FFFFFF'H,
              sspBitmask 'FF000000'H
            }
          },
          {
            psid 137,
            sspRange bitmapSspRange : {
              sspValue '01E0'H,
              sspBitmask 'FF1F'H
            }
          },
          {
            psid 138,
            sspRange bitmapSspRange : {
              sspValue '01C0'H,
              sspBitmask 'FF3F'H
            }
          },
```

```
            {
              psid 139,
              sspRange bitmapSspRange : {
                sspValue '01940000FFF8'H,
                sspBitmask 'FF0000000007'H
              }
            },
            {
              psid 140,
              sspRange bitmapSspRange : {
                sspValue '01FFFFE0'H,
                sspBitmask 'FF00001F'H
              }
            },
            {
              psid 141
            },
            {
              psid 623,
              sspRange bitmapSspRange : {
                sspValue '01C0'H,
                sspBitmask 'FF3F'H
              }
            }
          },
          minChainLength 2,
          eeType {app}
        },
        {
          subjectPermissions explicit : {
            {
              psid 623,
              sspRange bitmapSspRange : {
                sspValue '013E'H,
                sspBitmask 'FFC1'H
              }
            }
          }
        }
      },
      verifyKeyIndicator verificationKey : ecdsaBrainpoolP256r1 : compressed-y-1 :
'9ACFAF8ADEA9C44C205A09BB7C62C694DE3E4B97AEBF48B9A4D3A3A422BAECA0'H
    },
    signature ecdsaBrainpoolP256r1Signature : {
      rSig x-only :
'72FCEEA7A523D048A9126103D36679B823F8277439BDA9A797DA673E0988244E'H,
      sSig '4A9E4F88DD1AA1B90B6416736D097C71BC0BEB08A39A6C23C470E0E4AD9D48D5'H
    }
  }
```

Example of EA certificate

```
  EtsiTs103097Certificate ::= {
    version 3,
    type explicit,
    issuer sha256AndDigest : '35F33313404A473C'H,
    toBeSigned {
```

```
    id name : "BSI V2X Pilot PKI EA",
    cracaId '000000'H,
    crlSeries 0,
    validityPeriod {
      start 477243000,
      duration hours : 13128
    },
    appPermissions {
      {
        psid 623,
        ssp bitmapSsp : '010E'H
      }
    },
    certIssuePermissions {
        {
        subjectPermissions explicit : {
          {
            psid 623,
            sspRange bitmapSspRange : {
              sspValue '01C0'H,
              sspBitmask 'FF3F'H
            }
          }
        }
      }
    },
    encryptionKey {
      supportedSymmAlg aes128Ccm,
      publicKey eciesBrainpoolP256r1 : compressed-y-1 :
'A6EAC551C411D02C43B97B8F25F8B64155449D9F11E4E4855F5193FA9B12D910'H
    },
    verifyKeyIndicator verificationKey : ecdsaBrainpoolP256r1 : compressed-y-1 :
'93B2A4C9E1D1F434C3DDA1DF03D413A2040F110291C057F10849AAA1D5EA12ED'H
  },
  signature ecdsaBrainpoolP256r1Signature : {
    rSig x-only :
'29602DE33CF5E0B5222EB6ED9166692ECF493872C8AB7804119E054CBACA73C6'H,
    sSig '784FEC48A1E43E4C4071AF620DBBC34A7FB73480BE1835724BDB0CFBDCEA4B7A'H
  }
}
```

Example of AA certificate

```
EtsiTs103097Certificate ::= {
  version 3,
  type explicit,
  issuer sha256AndDigest : '35F33313404A473C'H,
  toBeSigned {
    id name : "BSI V2X Pilot PKI AA",
    cracaId '000000'H,
    crlSeries 0,
    validityPeriod {
      start 477243242,
      duration hours : 8760
    },
    appPermissions {
```

```
    {
      psid 623,
      ssp bitmapSsp : '0132'H
    }
  },
  certIssuePermissions {
    {
      subjectPermissions explicit : {
        {
          psid 36,
          sspRange bitmapSspRange : {
            sspValue '01FFFC'H,
            sspBitmask 'FF0003'H
          }
        },
        {
          psid 37,
          sspRange bitmapSspRange : {
            sspValue '01FFFFFF'H,
            sspBitmask 'FF000000'H
          }
        },
        {
          psid 137,
          sspRange bitmapSspRange : {
            sspValue '01E0'H,
            sspBitmask 'FF1F'H
          }
        },
        {
          psid 138,
          sspRange bitmapSspRange : {
            sspValue '01C0'H,
            sspBitmask 'FF3F'H
          }
        },
        {
          psid 139,
          sspRange bitmapSspRange : {
            sspValue '01940000FFF8'H,
            sspBitmask 'FF0000000007'H
          }
        },
        {
          psid 140,
          sspRange bitmapSspRange : {
            sspValue '01FFFFE0'H,
            sspBitmask 'FF00001F'H
          }
        },
        {
          psid 141
        }
      }
    }
  },
  encryptionKey {
    supportedSymmAlg aes128Ccm,
```

```
      publicKey eciesNistP256 : compressed-y-1 :
'7C6A29E10B28C6B5EDE509879096862BACA2B017CBAB304C16F12D173C81151D'H
    },
    verifyKeyIndicator verificationKey : ecdsaNistP256 : compressed-y-1 :
'D3432034D3D5C80F660076BEF8BC06306EA5D2E3A611B21B269B443918EFA29B'H
  },
  signature ecdsaBrainpoolP256r1Signature : {
    rSig x-only :
'6D4425E909516CDF1CCD204BCD865FA7807CD76A6DAD8FF670392918C8ECC29B'H,
    sSig '60A2701A5EBC9CD37BA3ED18C9B2BDF44F92D9EB22DAE58239AD52E1B7FA9042'H
  }
}
```

Example of RCA-CTL

```
RcaCertificateTrustListMessage ::= {
  protocolVersion 3,
  content signedData : {
    hashId sha256,
    tbsData {
      payload {
        data {
          protocolVersion 3,
          content unsecuredData : CONTAINING {
            version v1,
            content certificateTrustListRca : {
              version v1,
              nextUpdate 485016345,
              isFullCtl TRUE,
              ctlSequence 0,
              ctlCommands {
                add : ea : {
                  eaCertificate {
                  version 3,
                  type explicit,
                  issuer sha256AndDigest : '35F33313404A473C'H,
                  toBeSigned {
                    id name : "Test BSI V2X Pilot PKI EA",
                    cracaId '000000'H,
                    crlSeries 0,
                    validityPeriod {
                      start 477243000,
                      duration hours : 13128
                    },
                    appPermissions {
                      {
                        psid 623,
                        ssp bitmapSsp : '010E'H
                      }
                    },
                    certIssuePermissions {

                      {
                        subjectPermissions explicit : {
                          {
                            psid 623,
```

```
                                sspRange bitmapSspRange : {
                                  sspValue '01C0'H,
                                  sspBitmask 'FF3F'H
                                }
                              }
                            }
                          }
                        },
                        encryptionKey {
                          supportedSymmAlg aes128Ccm,
                          publicKey eciesBrainpoolP256r1 : compressed-y-1 :
'A6EAC551C411D02C43B97B8F25F8B64155449D9F11E4E4855F5193FA9B12D910'H
                        },
                        verifyKeyIndicator verificationKey : ecdsaBrainpoolP256r1 :
compressed-y-1 :
'93B2A4C9E1D1F434C3DDA1DF03D413A2040F110291C057F10849AAA1D5EA12ED'H
                      },
                      signature ecdsaBrainpoolP256r1Signature : {
                        rSig x-only :
'29602DE33CF5E0B5222EB6ED9166692ECF493872C8AB7804119E054CBACA73C6'H,
                        sSig
'784FEC48A1E43E4C4071AF620DBBC34A7FB73480BE1835724BDB0CFBDCEA4B7A'H
                      }
                    },
                    aaAccessPoint "http://test.bsi.v2x-pilot.escrypt.c" -- truncated
--,
                    itsAccessPoint "http://test.bsi.v2x-pilot.escrypt.c" -- truncated
--
                  },
                  add : aa : {
                    aaCertificate {
                      version 3,
                      type explicit,
                      issuer sha256AndDigest : '35F33313404A473C'H,
                      toBeSigned {
                        id name : "Test BSI V2X Pilot PKI AA",
                        cracaId '000000'H,
                        crlSeries 0,
                        validityPeriod {
                          start 477243242,
                          duration hours : 8760
                        },
                        appPermissions {
                          {
                            psid 623,
                            ssp bitmapSsp : '0132'H
                          }
                        },
                        certIssuePermissions {
                          {
                            subjectPermissions explicit : {
                              {
                                psid 36,
                                sspRange bitmapSspRange : {
                                  sspValue '01FFFC'H,
                                  sspBitmask 'FF0003'H
                                }
                              },
                              {
```

```
                                psid 37,
                                sspRange bitmapSspRange : {
                                  sspValue '01FFFFFF'H,
                                  sspBitmask 'FF000000'H
                                }
                              },
                              {
                                psid 137,
                                sspRange bitmapSspRange : {
                                  sspValue '01E0'H,
                                  sspBitmask 'FF1F'H
                                }
                              },
                              {
                                psid 138,
                                sspRange bitmapSspRange : {
                                  sspValue '01C0'H,
                                  sspBitmask 'FF3F'H
                                }
                              },
                              {
                                psid 139,
                                sspRange bitmapSspRange : {
                                  sspValue '01940000FFF8'H,
                                  sspBitmask 'FF0000000007'H
                                }
                              },
                              {
                                psid 140,
                                sspRange bitmapSspRange : {
                                  sspValue '01FFFFE0'H,
                                  sspBitmask 'FF00001F'H
                                }
                              },
                              {
                                psid 141
                              }
                            }
                          }
                        },
                        encryptionKey {
                          supportedSymmAlg aes128Ccm,
                          publicKey eciesNistP256 : compressed-y-1 :
'7C6A29E10B28C6B5EDE509879096862BACA2B017CBAB304C16F12D173C81151D'H
                        },
                        verifyKeyIndicator verificationKey : ecdsaNistP256 :
compressed-y-1 :
'D3432034D3D5C80F660076BEF8BC06306EA5D2E3A611B21B269B443918EFA29B'H
                      },
                      signature ecdsaBrainpoolP256r1Signature : {
                        rSig x-only :
'6D4425E909516CDF1CCD204BCD865FA7807CD76A6DAD8FF670392918C8ECC29B'H,
                        sSig
'60A2701A5EBC9CD37BA3ED18C9B2BDF44F92D9EB22DAE58239AD52E1B7FA9042'H
                      }
                    },
                    accessPoint "http://test.bsi.v2x-pilot.escrypt.c" -- truncated --
                  },
                  add : dc : {
```

```
                url "http://test.bsi.v2x-pilot.escrypt.c" -- truncated --,
                cert {
                  '35F33313404A473C'H
                }
              }
            }
          }
        }
      }
    },
    headerInfo {
      psid 624,
      generationTime 477240345929000
    }
  },
  signer certificate : {
    {
      version 3,
      type explicit,
      issuer self : sha256,
      toBeSigned {
        id name : "Test BSI V2X Pilot PKI Root",
        cracaId '000000'H,
        crlSeries 0,
        validityPeriod {
          start 477215871,
          duration hours : 17544
        },
        appPermissions {
          {
            psid 624,
            ssp bitmapSsp : '0138'H
          },
          {
            psid 622,
            ssp bitmapSsp : '01'H
          }
        },
        certIssuePermissions {
          {
            subjectPermissions explicit : {
              {
                psid 36,
                sspRange bitmapSspRange : {
                  sspValue '01FFFC'H,
                  sspBitmask 'FF0003'H
                }
              },
              {
                psid 37,
                sspRange bitmapSspRange : {
                  sspValue '01FFFFFF'H,
                  sspBitmask 'FF000000'H
                }
              },
              {
                psid 137,
                sspRange bitmapSspRange : {
                  sspValue '01E0'H,
```

```
                    sspBitmask 'FF1F'H
                  }
                },
                {
                  psid 138,
                  sspRange bitmapSspRange : {
                    sspValue '01C0'H,
                    sspBitmask 'FF3F'H
                  }
                },
                {
                  psid 139,
                  sspRange bitmapSspRange : {
                    sspValue '01940000FFF8'H,
                    sspBitmask 'FF0000000007'H
                  }
                },
                {
                  psid 140,
                  sspRange bitmapSspRange : {
                    sspValue '01FFFFE0'H,
                    sspBitmask 'FF00001F'H
                  }
                },
                {
                  psid 141
                },
                {
                  psid 623,
                  sspRange bitmapSspRange : {
                    sspValue '01C0'H,
                    sspBitmask 'FF3F'H
                  }
                }
              },
              minChainLength 2,
              eeType {app}
            },
            {
              subjectPermissions explicit : {
                {
                  psid 623,
                  sspRange bitmapSspRange : {
                    sspValue '013E'H,
                    sspBitmask 'FFC1'H
                  }
                }
              }
            }
          },
          verifyKeyIndicator verificationKey : ecdsaBrainpoolP256r1 : compressed-y-
1 : '9ACFAF8ADEA9C44C205A09BB7C62C694DE3E4B97AEBF48B9A4D3A3A422BAECA0'H
        },
        signature ecdsaBrainpoolP256r1Signature : {
          rSig x-only :
'72FCEEA7A523D048A9126103D36679B823F8277439BDA9A797DA673E0988244E'H,
          sSig '4A9E4F88DD1AA1B90B6416736D097C71BC0BEB08A39A6C23C470E0E4AD9D48D5'H
        }
      }
```

```
    },
    signature ecdsaBrainpoolP256r1Signature : {
      rSig x-only :
'1D537B5E5A2FEFAF2FC12CF7E2AAAD98A3B699E7C2707719C4F27BFEB785A8BE'H,
      sSig '9EAFE59AD1F51FF122F16E988D70B6286987EBE93F57F3B0BF1F1072B2543544'H
    }
  }
}
```

Example of CRL

```
CertificateRevocationListMessage ::= {
  protocolVersion 3,
  content signedData : {
    hashId sha256,
    tbsData {
      payload {
        data {
          protocolVersion 3,
          content unsecuredData : CONTAINING {
            version v1,
            content certificateRevocationList : {
              version v1,
              thisUpdate 477843763,
              nextUpdate 485619763,
              entries {
                '7044E6B91D9E3DC6'H
              }
            }
          }
        }
      },
      headerInfo {
        psid 622,
        generationTime 477843763856000
      }
    },
    signer certificate : {
      {
        version 3,
        type explicit,
        issuer self : sha256,
        toBeSigned {
          id name : "Test BSI V2X Pilot PKI Root",
          cracaId '000000'H,
          crlSeries 0,
          validityPeriod {
            start 477215871,
            duration hours : 17544
          },
          appPermissions {
            {
              psid 624,
              ssp bitmapSsp : '0138'H
            },
```

```
    {
      psid 622,
      ssp bitmapSsp : '01'H
    }
  },
  certIssuePermissions {
    {
      subjectPermissions explicit : {
        {
          psid 36,
          sspRange bitmapSspRange : {
            sspValue '01FFFC'H,
            sspBitmask 'FF0003'H
          }
        },
        {
          psid 37,
          sspRange bitmapSspRange : {
            sspValue '01FFFFFF'H,
            sspBitmask 'FF000000'H
          }
        },
        {
          psid 137,
          sspRange bitmapSspRange : {
            sspValue '01E0'H,
            sspBitmask 'FF1F'H
          }
        },
        {
          psid 138,
          sspRange bitmapSspRange : {
            sspValue '01C0'H,
            sspBitmask 'FF3F'H
          }
        },
        {
          psid 139,
          sspRange bitmapSspRange : {
            sspValue '01940000FFF8'H,
            sspBitmask 'FF0000000007'H
          }
        },
        {
          psid 140,
          sspRange bitmapSspRange : {
            sspValue '01FFFFE0'H,
            sspBitmask 'FF00001F'H
          }
        },
        {
          psid 141
        },
        {
          psid 623,
          sspRange bitmapSspRange : {
            sspValue '01C0'H,
            sspBitmask 'FF3F'H
          }
```

```
                    }
                  },
                  minChainLength 2,
                  eeType {app}
                },
                {
                  subjectPermissions explicit : {
                    {
                      psid 623,
                      sspRange bitmapSspRange : {
                        sspValue '013E'H,
                        sspBitmask 'FFC1'H
                      }
                    }
                  }
                }
              },
              verifyKeyIndicator verificationKey : ecdsaBrainpoolP256r1 : compressed-y-
    1 : '9ACFAF8ADEA9C44C205A09BB7C62C694DE3E4B97AEBF48B9A4D3A3A422BAECA0'H
            },
            signature ecdsaBrainpoolP256r1Signature : {
              rSig x-only :
    '72FCEEA7A523D048A9126103D36679B823F8277439BDA9A797DA673E0988244E'H,
              sSig '4A9E4F88DD1AA1B90B6416736D097C71BC0BEB08A39A6C23C470E0E4AD9D48D5'H
            }
          }
        },
        signature ecdsaBrainpoolP256r1Signature : {
          rSig x-only :
    'A54E0C59CE3AD8C5D9660112C8546F6EF853FE837D2901F64ABB7FA4A23DD0DD'H,
          sSig '14EF3E9BA921537CAFDDA39FFEA81E4A1A2F979FFAD7C69E194AF2F9D4B0E543'H
        }
      }
}
```